Estimation of Snow Depth in Northern Sweden: Using Gaussian Markov Random Fields

Magnus Öhlund

Master's thesis
2013:E19

Lund University

Faculty of Engineering
Centre for Mathematical Sciences
Mathematical Statistics

Centrum Scientiarum Mathematicarum

ABSTRACT

The goal of this dissertation is to estimate the amount of water in snowpacks at the areas Kultsjön and Rensaren in northen Sweden. A model is built using spatial statistics, and the estimations will be done by Integrated Nested Laplace Approximations (INLA). Then a validation process was done by removing data from the data set to create a validation set, and compare the estimations done by INLA to a simple linear regression. Finally the model is applied to the areas of interest, and the results are analysed.

ACKNOWLEDGEMENTS

I'd like to thank my supervisor Johan Lindström from the Centre for Mathematical sciences, Lund university for all his help during this project. I also would like to thank David Gustafsson from KTH for the data used in this dissertation.

CONTENTS

Part I

INTRODUCTION

1 BACKGROUND

When trying to predict the spring and summer inflow of water into reservoirs of hydroelectric power stations, estimation of snow depth in mountain regions of northen Sweden is of great importance.

Using an ultrasound receiver, towed behind a snowmobile, high quality measurements of the snow depth along lines have been obtained, and will be the subject of modelling. The goal of this Master's dissertation is to develop a model, using the availible data, to estimate the snow depth at unobserved locations.

The dissertation starts with presenting the availible data in chapter 2. The theory behind our model is then presented in chapter 3, and then applied to our data to build our model in chapter 4. The modelling is based on spatial statistcs (see Handbook in Spatial Statistics [6]), Gaussian Markov Random Fields, and Integrated Nested Laplace Approximations.

Some validation on the estimations are performed in chapter 5, where we create some test data to validate on, and then apply the model to estimate on the area of interest; we discuss the results, together with some suggestions of future work in chapter 6.

Plenty of previous works done on snow estimation exists; the papers used in the modelling process used a binary regression tree (Winstrahl et al. [19]), and kriging with a nonlinear trend model (Erickson and Williams [4]) were used. To our knowledge, no previous works on snow depth using Integrated Nested Laplace Approximations (see `www.r-inla.org`) have been done.

2 DATA

The availible data is: elevation, wind shelter, density of the snow, and snow depth. The elevation, and wind shelter data exists on a grid with 50 metres between grid points and will be used as covariates in the modelling process.

The density of the snow, and snow depth were measured at locations typically located between the grid points. The density and depth of the snow is used to calculate the amount of water at each observed location (more on this later on in 2.2).

Because most of the observations were located in-between grid points, an interpolation is applied to obtain covariate data at observation locations. Matlab's `interp2` is used with a bilinear interpolation option to accomplish this (see 3.1).

2.1 AREA

The areas Kultsjön and Rensaren together creates the area to be modelled in this project. A map of the surrounding area together with the snow measures can be seen in Figure 1. Each point on this map has elevation, and wind shelter data available.

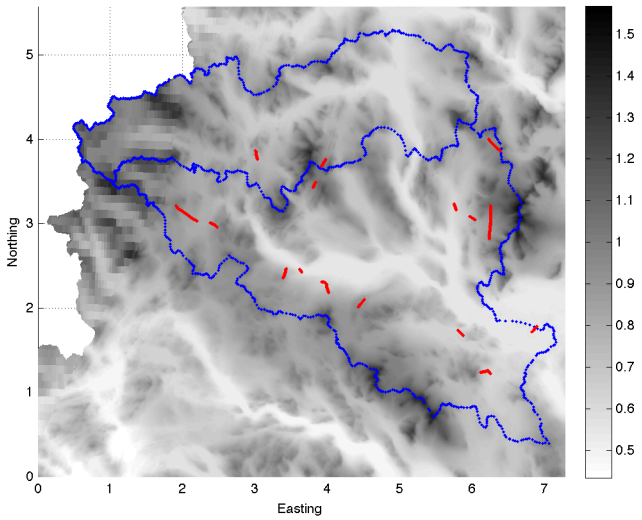


Figure 1: Picture of the map. The blue line outlines the area of interest, and the red lines are the observations.

In the upper-left corner of the area of interest, many neighbouring points shares the same elevation, which implies that the area has

lower resolution than other areas on the map. Because several points at this area shares the same height, the slope in this area can not be correctly calculated. This causes the wind shelter data calculation (relies on slope, as shown in 2.4) to fail, and will cause strange values in this area.

Cutting off these points, with the surrounding area (which we are not interested in), the grid to be modelled can be seen in Figure 3. Note that we will have to apply bilinear interpolation before removing the low-resolution area, since one of the observation lines reaches into the area to be removed. We will have to ensure reasonable covariates along this line.

Plots showing the covariates of the line extending into the low resolution region can be seen in Figure 2. The elevation of the line inside the low resoluted area has a quite steep rise, but a rise of about 140 metres in about 1.2 km is not entirely unreasonable. While the wind shelter data seems a bit strange, the values vary within a reasonable range.

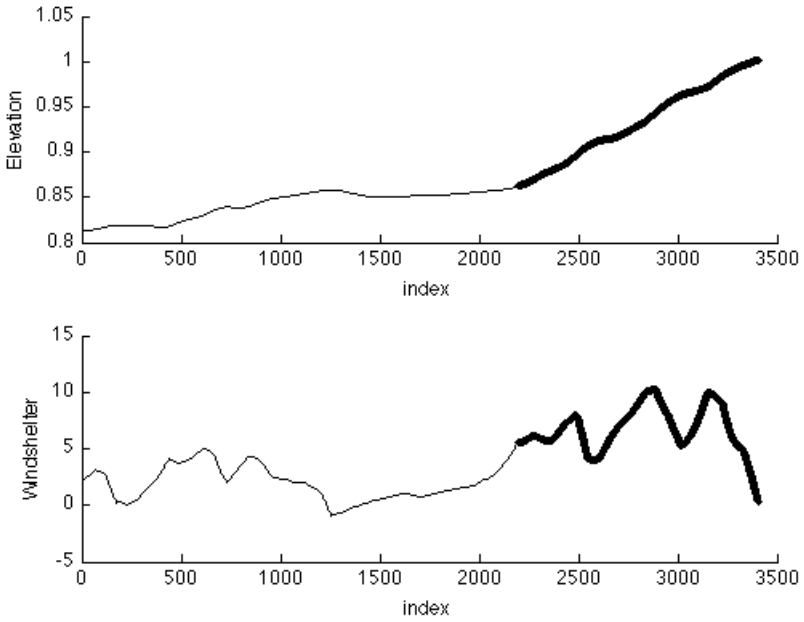


Figure 2: Elevation and wind shelter data of the line extending into the removed area. The bold part is inside the area with low resolution.

Since the covariate data of the line doesn't seem to be too unreasonable, it will be kept in the model.

2.2 SNOW WATER EQUIVALENT

Snow Water Equivalent (SWE) is a measure of the water content in a pack of snow; it can be interpreted as the resulting water depth if all

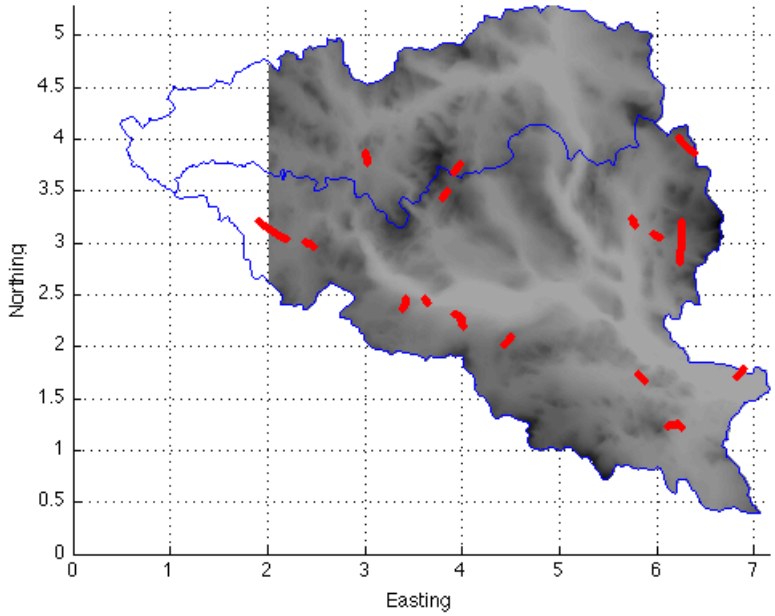


Figure 3: Map of the area with the lines of observations. The area with low resolution is removed together with the surrounding areas. Note the line stretching out into the removed area.

snow would melt. SWE is measured in metres as a function of snow depth, water density, and snow density in this project; it can also be represented as kg/m^2 (NASA [14]).

Definition 1. *Snow Water Equivalent (SWE)*

$$SWE = \text{snow depth} \times \frac{\text{snow density}}{\text{water density}} \tag{2.1}$$

The observations of SWE data will be modelled using Gaussian distributions 3.2). If the data has significant skewness any analysis based on Gaussian distributions will provide incorrect estimates (Morel [13]) and a Gaussian model would be inappropriate. A histogram of the SWE data of the lines can be seen in the upper left graph in Figure 4, where it is easy to see a long right tail of the SWE data. A transformation will have to be applied to reduce skewness of the data in order to improve model fit.

Three simple transformations are tested: $\sqrt{SWE}$, $\sqrt[3]{SWE}$, and $\log(SWE+1)$ (Box and Cox [2]), where the skewness is estimated by $b_1 = (\frac{n-1}{n})^{3/2}\frac{\mu_3}{\mu_2^{3/2}}$ (Gill and Joanes [7]).

The original data had a skewness estimation of ≈ 1.74, the $\sqrt{SWE}$ transformation ≈ 0.78, $\sqrt[3]{SWE} \approx 0.43$, and the tranformation $\log(SWE+1)$ yielded the estimation ≈ -0.36. Graphs of the transformations can be seen in Figure 4.

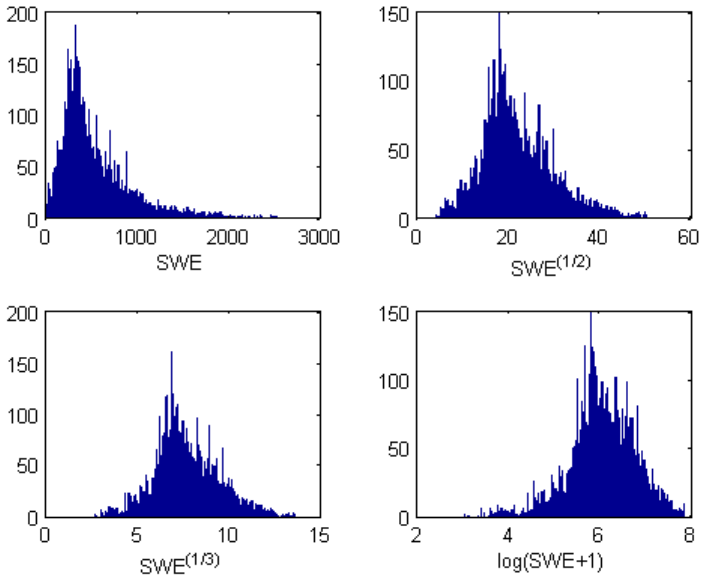


Figure 4: Application of the transformations discussed in 2.2.

The log transform gave skewness closest to zero, and will be used in the modelling. Note that some skewness remains, and could perhaps be improved on.

2.3 ELEVATION

The elevation is measured in kilometers, and is obtained from the map of the region. A summary of the elevation data can be found in the table in Table 1. The range of elevation data in the lines is smaller than for the entire area, and we need to be careful with predictions that extrapolate to unobserved elevation values.

2.4 WIND SHELTER

As suggested by Winstrahl et al. [19], a way of characterising the effect of wind on snow could be a significant factor when predicting snow depth. The wind shelter factor tries to quantify the degree of shelter/exposure and thus quantify where snow gathers. It is calculated as:

Definition 2. *Wind shelter*
For the search distance dx*, the wind shelter is calculated as*

$$S_{dx}(\mathbf{u}) = \max_{\mathbf{u}_v} \left[\tan\left(\frac{H(\mathbf{u}_v) - H(\mathbf{u})}{|\mathbf{u}_v - \mathbf{u}|} \right) \right], \tag{2.2}$$

where H *is the elevation of point* $\mathbf{u}$*, and* $\mathbf{u}_v$ *is a point within the search distance in the chosen wind direction* $\mathbf{u}_v = \mathbf{u} + v \cdot t, 0 \leqslant t \leqslant dx$.

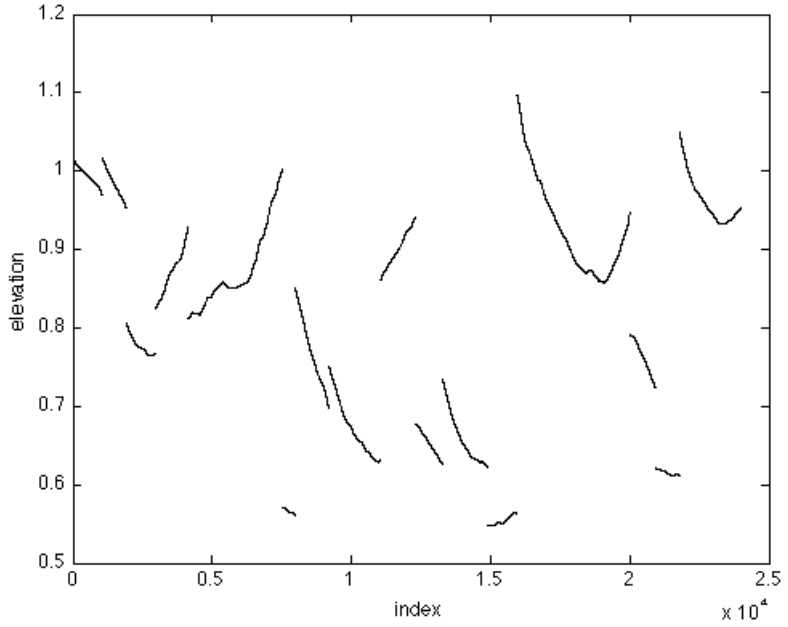


Figure 5: The elevation of the lines, each segment is a single line.

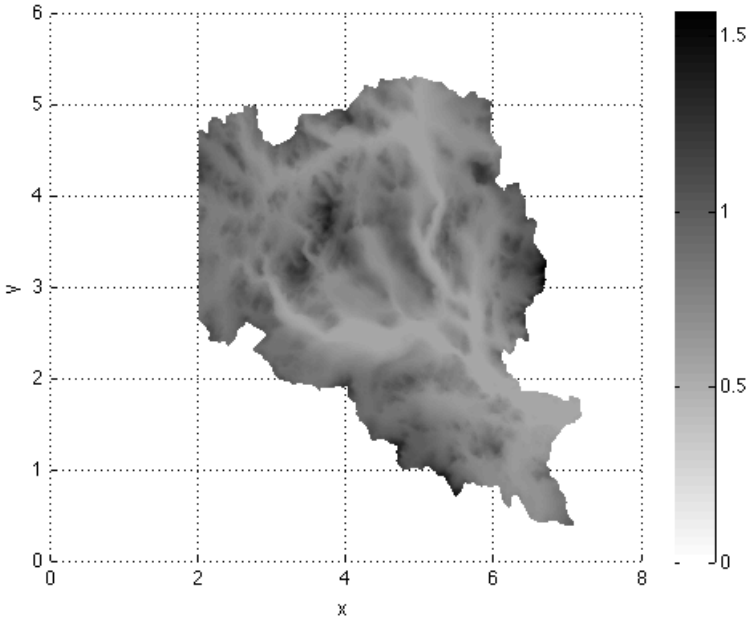


Figure 6: Elevation in km of the area of focus.

The wind shelter data was computed for eight different wind directions; 22.5°, 67.5°, 112.5°, 157.5°, 202.5°, 247.5°, 292.5°, 337.5°. For each direction, wind shelter has been computed using three different seach distances; 250, 750, and 1500 metres.

As with the elevation data, a summary of wind shelter data can be found in Table 1. The table shows that the representation of the areas

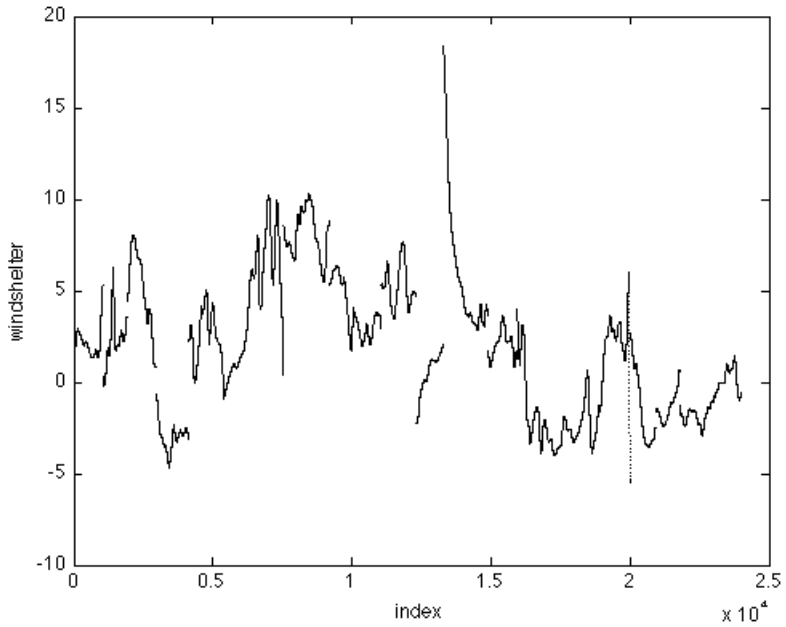


Figure 7: Wind shelter of the lines.

wind shelter is quite poor in the measurement locations. Unlike the elevation data, where the area data was only slightly outside the line data, the range of wind shelter data for the entire area is a magnitude larger than for the observations.

Using all wind shelter data in the model would be infeasible. Erickson and Williams [4], and Winstrahl et al. [19] only used wind shelter for the dominant wind direction and our aim is to do the same here. However, the dominant wind direction is unknown, we also have three different search distances. Determining which combination of wind direction and search distance to use will be done in 4.1.2.

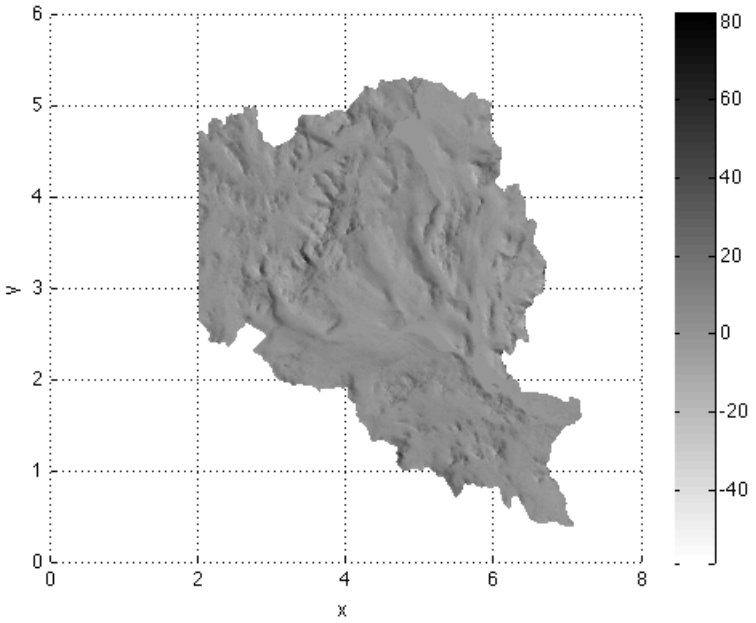


Figure 8: Wind shelter of the area.

	Observations			Grid		
	min	mean	max	min	mean	max
elv	0.548	0.820	1.10	0.536	0.790	1.57
wsh	-5.49	2.07	18.4	-38.8	2.85	61.0
SWE	20.2	534	2520	—	—	—
log(SWE + 1)	3.05	6.06	7.83	—	—	—
x	2.01	4.55	7.15	1.91	4.66	6.89
y	0.395	3.09	5.29	1.22	2.83	4.00
elv.rm	—	—	—	0.803	1.02	1.40
wsh.rm	—	—	—	-58.9	6.62	82.0

Table 1: Wind shelter is abbreviated as wsh, elevation as elv, and the suffix .rm refers to the removed data of the area with low resolution. x and y are the coordinates on the map.

3

THEORY

This chapter presents the theory used in the modelling.

3.1 INTERPOLATION

Since the observations are located in-between grid points, we need to interpolate from the grid points to create covariate data at the observation locations. Bilinear interpolation is a interpolation technique used to find the value of a point given four surrounding points. One could think of it as linear interpolations in three dimensions (Farin [5, chap. 14]).

Definition 3. *Linear interpolation*
Let $\mathbf{u}_1$ *and* $\mathbf{u}_2$ *be two points on* $\mathbb{R}^2$*, then if the* x*-coordinate of a point* $\mathbf{u}$ *between these points is known, the coordinates of* $\mathbf{u}$ *can be calculated as*

$$\mathbf{u} = t\mathbf{u}_1 + (1-t)\mathbf{u}_2, \; t \in [0,1]. \tag{3.1}$$

where $t = \frac{x_2 - x}{x_2 - x_1}$*;* (x_i, y_i)*,* $i = 1, 2$ *is the coordinates of* $\mathbf{u}_i$*.*

This can be seen as a weighted average, with the weights $\frac{x_2-x}{x_2-x_1}$ and $\frac{x-x_1}{x_2-x_1}$.

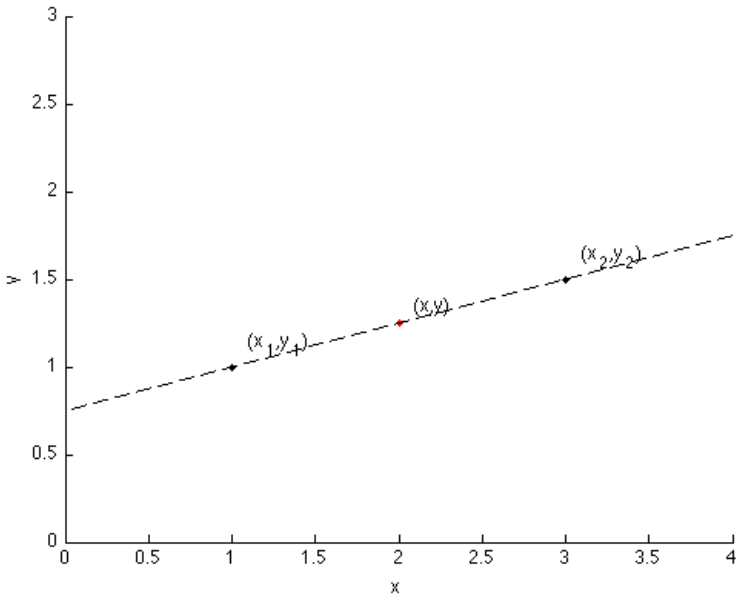


Figure 9: Example of linear interpolation with points $\mathbf{u}_1 = (x_1, y_1)$, $\mathbf{u}_2 = (x_2, y_2)$ to a point $\mathbf{u} = (x, y)$.

Now assume four points $\mathbf{u}_{1,1}$, $\mathbf{u}_{1,2}$,$\mathbf{u}_{2,1}$,$\mathbf{u}_{2,2}$ in $\mathbb{R}^3$. The bilinear interpolation is done by first interpolating in direction t

$$\begin{aligned}\mathbf{u}_1 &= t\mathbf{u}_{1,1} + (1-t)\mathbf{u}_{1,2},\\ \mathbf{u}_2 &= t\mathbf{u}_{2,1} + (1-t)\mathbf{u}_{2,2},\end{aligned} \tag{3.2}$$

and then interpolated again in the other direction s (note that it does not matter which direction we start in).

$$\mathbf{u} = s\mathbf{u}_1 + (1-s)\mathbf{u}_2, \tag{3.3}$$

and we have interpolated the four points $\mathbf{u}_{1,1}$, $\mathbf{u}_{2,1}$,$\mathbf{u}_{1,2}$,$\mathbf{u}_{2,2}$ to a point $\mathbf{u}$. The interpolation process can be written in one step as:

$$\mathbf{u} = s\left[t\mathbf{u}_{1,1} + (1-t)\mathbf{u}_{1,2}\right] + (1-s)\left[t\mathbf{u}_{2,1} + (1-t)\mathbf{u}_{2,2}\right], \tag{3.4}$$

where $t = \frac{x_2-x}{x_2-x_1}$, and $s = \frac{y_2-y}{y_2-y_1}$.

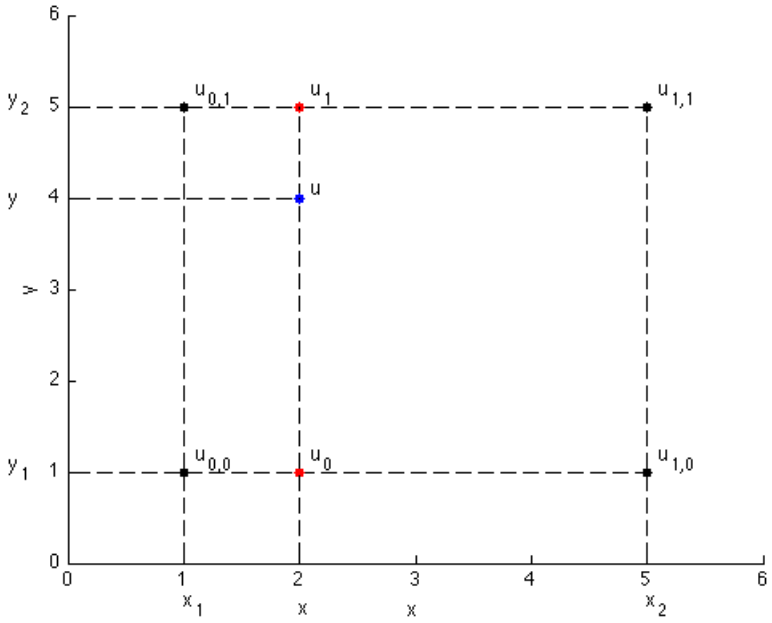


Figure 10: Graphic example of bilinear interpolation

3.2 SPATIAL MODELLING

The common way to model a spatial stochastic process in geostatistics consists of a latent Gaussian field, $\mathbf{x}$, modelled as a normal distribution with mean μ and covariance $\mathbf{\Sigma}$ as

$$\mathbf{x} \in \mathcal{N}(\mu, \mathbf{\Sigma}). \tag{3.5}$$

Observing the latent field $\mathbf{x}$ at locations $\mathbf{u}$ we can form a new distribution of these observations. The observations $\mathbf{y}$ of the latent field $\mathbf{x}$ are also modelled as a normal distribution

$$\mathbf{y}|\mathbf{x} \in \mathcal{N}(\mathbf{A}\mathbf{x}, \sigma_\epsilon^2), \tag{3.6}$$

where $\mathbf{A}$ is a matrix extracting appropriate elements from $\mathbf{x}$, and σ_ϵ^2 is the nugget effect [6, chap. 2].

Definition 4. *Nugget effect*
The nugget effect ε at the point $\mathbf{u}$ is spatially uncorrelated error that describes the (observational) error in repeated observations at $\mathbf{u}$. The nugget has zero mean and variance σ_ϵ^2.

The model parameter(s) are then added to form the following model for the observations and latent field:

$$\begin{aligned} \mathbf{x}|\theta &\in \mathcal{N}(\mu, \Sigma(\theta)), \\ \mathbf{y}|\mathbf{x}, \theta &\in \mathcal{N}(\mathbf{A}\mathbf{x}, \sigma_\epsilon^2), \end{aligned} \tag{3.7}$$

where $\Sigma(\theta)$ is a parametric model for the covariance of $\mathbf{x}$, with the parameters θ; the Matérn covariance model will be used in this project (see 3.2.1). At a observation location $\mathbf{u}$, the observation $\mathbf{y}$ can be modelled as a sum of parts:

$$y(\mathbf{u}) = \mu(\mathbf{u}) + \eta(\mathbf{u}) + \epsilon(\mathbf{u}), \tag{3.8}$$

where μ is the mean component, η is the field spatial component described by $\Sigma(\theta)$, and ϵ is the nugget effect. The mean component can be modelled using a parametric mean model of y, $\mu(\mathbf{u}) = E[y(\mathbf{u})]$; the spatial component $\eta(\mathbf{u})$ tries to model the remaining spatial structure, see further 3.3.

The goal of the modelling process is to use the observations $\mathbf{y}$ to find the distribution of the unknown parameters $\pi(\theta|\mathbf{y})$, and to reconstruct the latent field $\mathbf{x}$, $\pi(\mathbf{x}|\mathbf{y})$.

3.2.1 *Random field*

A simple interpertation of a random field is describing it as a vector of stochastic variables, with a joint distribution[6, chap. 2]. However, a more formal definition is:

Definition 5. *Random field*
A random field $\tilde{x}(u), u \in \Omega$, is a random function defined on some index set Ω.

Where the index set typically is $\Omega \subseteq \mathbb{R}^2$ or $\Omega \subseteq \mathbb{N}^2$. A random field with a Gassian probability density is called a Gaussian Random Field (GRF).

To calculate the covariance between two points on a GRF, separated by a distance d, on a random field, a parametric model for the covariance is used. The model used in this dissertation is the Matérn covariance, which is a class of isotropic covariance functions named after Bertil Matérn [6, chap. 5].

Definition 6. *Matérn covariance*
The Matérn covariance for two points, d *distance units apart is given by*

$$r_M(d) = \sigma^2 \frac{1}{\Gamma(\nu)2^{\nu-1}} (\mathcal{K}d)^\nu \mathbf{K}_\nu(\mathcal{K}d), \tag{3.9}$$

where Γ *is the gamma function,* $\mathcal{K}$ *is a scale parameter,* ν *is a shape parameter, and* $\mathbf{K}_\nu$ *is the modified Bessel function of the second kind.*

The range parameter can be calculated with the empirically derived definition $\rho = \sqrt{8\nu}/\kappa$ [11].

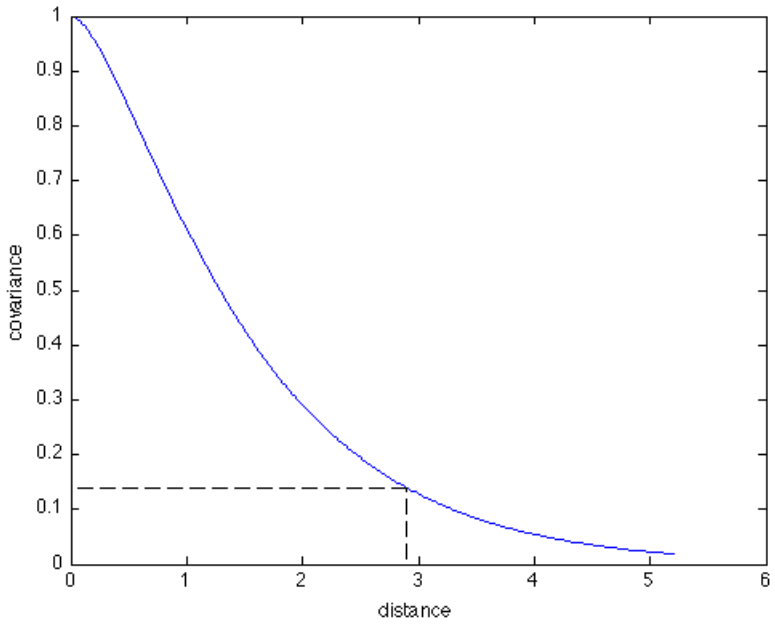


Figure 11: Example of Matérn covariance with $\sigma^2 = 1, \nu = 1$, $\kappa \approx 0.977$, resulting in range $\rho \approx 2.89$.

3.3 GAUSSIAN RANDOM MARKOV FIELD

A major issue with Gaussian fields for huge datasets is the big and dense covariance matrix of the latent field $\mathbf{x}$ (3.5), which leads to expensive computations. With a GMRF, we use the precision matrix $\mathbf{Q} = \mathbf{\Sigma}^{-1}$, which is constructed using the markov property for some neighbourhood structure $\mathcal{N}$. This leads to a $\mathbf{Q}$ with sparse structure, which sparsity makes it possible to achieve more efficent computations.

3.3.1 *Markov property*

To define the Markov property for the Gaussian Random Field (a random field with a Gaussian probability density), a neighbourhood

structure is required to describe the range of conditioned dependence between locations.

Definition 7. *Neighbourhood structure*
A neighbour structure is a symmetric relation which defines which locations are neighbours of each other (Lindgren [10, chap. 4]).

$$\begin{aligned} \mathcal{N}(\mathbf{u}) &= \{\boldsymbol{\nu};\ \text{such that } \boldsymbol{\nu} \text{ is a neighbour of } \mathbf{u}\}, \\ \boldsymbol{\nu} \in \mathcal{N}(\mathbf{u}) &\Leftrightarrow \mathbf{u} \in \mathcal{N}(\boldsymbol{\nu}) \end{aligned} \tag{3.10}$$

The neighbourhood structure can be used to define a Markov property, which puts requirements on the conditional distribution of one location given the value of all other locations.

Definition 8. *The Markov condition*

$$P(\tilde{x}(\mathbf{u}) = x \mid \tilde{x}(\boldsymbol{\nu}),\ \boldsymbol{\nu} \neq \mathbf{u}) = P(\tilde{x}(\mathbf{u}) = x \mid \tilde{x}(\boldsymbol{\nu}),\ \boldsymbol{\nu} \in \mathcal{N}(\mathbf{u})) \tag{3.11}$$

Under the Markov condition, the conditional distribution of $\tilde{x}(\mathbf{u})$ depends on the rest of the field only through its neighbours [10, chap. 4]. The implication for the precision matrix $\mathbf{Q}$ will be that for each location $\mathbf{u}$, only the neighbours of $\mathbf{u}$ have $\mathbf{Q}_{u,v} \neq 0$ in the precision matrix, with zeros for all non-neighbouring points. The markov condition creates the sparsity structure for the precision matrix $\mathbf{Q}$, therefore avoiding the dense covariance matrix $\boldsymbol{\Sigma}$, leading to computational gains.

The simplest example of a GMRF is an AR(1)-process in time series analysis (Madsen [12]); like an AR(1)-process conditionally depending on only the nearest previous value in time, so for a GMRF the conditional distribution of $\mathbf{u}$ depends on only its spatial neighbours, $\mathcal{N}_{\mathbf{u}}$.

3.3.2 *Definition of a GMRF*

Definition 9. *Gaussian Markov Random Field (GMRF)*
A Gaussian random field $\mathbf{x} \in \mathbb{R}^n$ is called a GMRF with mean $\boldsymbol{\mu}$ and precision matrix $\mathbf{Q} > 0$, iff its probability density has the form

$$\pi(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}} |\mathbf{Q}|^{1/2} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \mathbf{Q}(\mathbf{x}-\boldsymbol{\mu})\right), \tag{3.12}$$

and satisfies the markov property (3.11) for some neighbourhood $\mathcal{N}$.

3.3.3 *Stochastic Partial Differential Equations*

Stochastic Differential Equations (SDE) are a class of differential equations where the driving right hand side is replaced by stochastic noise; as an extension, a Stochastic Partial Differential Equation (SPDE) is a

SDE of more than one parameter (Holden et al. [8]). No theory regarding SPDEs is presented here, as such theory is beyond the scope of this paper; only a brief overview will be presented here.

According to Whittle [17, 18], fields in $\mathbb{R}^d$ with Matérn covariances are solutions to the following SPDE:

$$(\kappa - \Delta)^{\alpha/2}\tau x(\mathbf{u}) = \mathcal{W}(\mathbf{u}), \quad \alpha = \nu + d/2, \tag{3.13}$$

where $\mathcal{W}$ is Gaussian white noise, τ is variance scaling, and Δ is the Laplacian $\Delta = \sum_{i=1}^{n} \frac{\partial}{\partial x_i^2}$; ν and κ from (3.9).

In Lindgren et al. [11] it is shown how the SPDE (3.13) can be solved using a finite element method on a triangulation of the data locations (see 4.2.1). The solution can then be constructed as a basis function expansion

$$x(\mathbf{u}) = \sum_{k=1}^{n} \psi_k(\mathbf{u}) w_k, \tag{3.14}$$

with Gaussian-distributed weights w_k, and basis functions ψ_k. The basis functions are chosen to be piecewise linear in each triangle, such that $\psi_k = 1$ at vertex k, and 0 at all other vertrices [11].

The precision matrix of the weights in (3.14), $\mathbf{Q}_{\alpha,\kappa^2}$, can be written as:

$$\begin{aligned} \mathbf{Q}_{1,\kappa^2} &= \mathbf{K}_{\kappa^2} \\ \mathbf{Q}_{2,\kappa^2} &= \mathbf{K}_{\kappa^2}\mathbf{C}^{-1}\mathbf{K}_{\kappa^2} \\ \mathbf{Q}_{\alpha,\kappa^2} &= \mathbf{K}_{\kappa^2}\mathbf{C}^{-1}\mathbf{Q}_{\alpha-2,\kappa^2}\mathbf{K}_{\kappa^2}\mathbf{C}^{-1}, \ \alpha = 3, 4, \dots \end{aligned} \tag{3.15}$$

Where the $n \times n$ matrices $\mathbf{C}$,$\mathbf{G}$, and $\mathbf{K}$ have entries

$$\begin{aligned} C_{ij} &= < \psi_i, \psi_j > \\ G_{ij} &= < \nabla\psi_i, \nabla\psi_j > \\ \mathbf{K}_{\kappa^2} &= \kappa^2\mathbf{C} + \mathbf{G}, \end{aligned} \tag{3.16}$$

where $< f, g >$ is the scalar product $< f, g >= \int fg \, dx$

3.4 BAYESIAN STATISTICS

In classical (or frequentist) statistics, data would be treated as a random sample generated from a distribution with fixed parameters; whereas in Bayesian statistics we assume that data is fixed and treat the parameters as unobserved random variables (Kypraios [9]). Central to Bayesian statistics (hence its name) is Bayes' theorem.

Definition 10. *Bayes' theorem*

$$\pi(\theta|\mathbf{y}) = \frac{\pi(\mathbf{y}|\theta)\pi(\theta)}{\pi(\mathbf{y})} \propto \pi(\mathbf{y}|\theta)\pi(\theta). \tag{3.17}$$

As described in 3.2, we let θ denote the parameters, and y the observed data.

- $\pi(\theta)$ is the prior distribution, which summaries the prior knowledge of the parameters θ.
- $\pi(y|\theta)$ likelihood function, how probable it is to have observed y given θ.
- The posterior distribution $\pi(\theta|y)$ summarises the information about θ given by data and priors.
- $\pi(y)$ is the marginal distribution, and is often treated as an unknown (and ignorable) normalising constant in the posterior.

3.4.1 *Prediction and estimation*

Continuing from 3.2, where our goal is to predict the posterior distrubition of the latent field x, given the observations y, $\pi(x|y)$, and estimate the posterior of the parameters θ, given the observations y, $\pi(\theta|y)$. Our observational data y are measurements of a latent field x, such that y is a conditional distribution given x and θ. Using Bayes' theorem (10) we can write these two posteriors as

$$\pi(\theta|y) = \frac{\pi(y|\theta)\pi(\theta)}{\pi(y)} \propto \pi(y|\theta)\pi(\theta) \tag{3.18}$$

$$\pi(x|y,\theta) = \frac{\pi(y|x,\theta)\pi(x|\theta)\pi(\theta)}{\pi(y,\theta)} \propto \pi(y|x,\theta)\pi(x|\theta)\pi(\theta). \tag{3.19}$$

Using (3.18), we estimate the posterior of the parameters, given the observations, as:

$$\pi(\theta|y) = \int \pi(x,y|\theta)\pi(\theta)\mathrm{d}x = \int \pi(y|x,\theta)\pi(x|\theta)\pi(\theta)\mathrm{d}x, \tag{3.20}$$

The posterior distribution of the field, given the observations, can be obtained by integrating out the parameter uncertainty in (3.19)

$$\pi(x|y) = \int \pi(x,\theta|y)\mathrm{d}\theta = \int \pi(x|y,\theta)\pi(\theta|y)\mathrm{d}x. \tag{3.21}$$

Usually these integrals are too hard to solve analytically, except for rather simple models (Sköld [16]). (3.7) is an example of a simpler model, because x and $y|x$ are both Gaussian. However, $\pi(\theta|y)$ in (3.21) is an unknown distribution, and the integral (3.21) will have to be computed using numerical methods e.g. Monte-Carlo methods, or Integrated Nested Laplace Approximations, which will be described in 3.5.

3.5 INTEGRATED NESTED LAPLACE APPROXIMATIONS

A short introduction to Integrated Nested Laplace Approximations (INLA) is presented in this section. INLA is an alternative to using Markov Chain Monte Carlo (MCMC) [15] to find a numerical solution to the integrals (3.20) and (3.21).

INLA's approach to Bayesian inference is as follows: the latent field $\mathbf{x}$ is assumed to be GMRF, and our distribution of observations $\pi(\mathbf{y}|\mathbf{x},\theta)$ factors as $\pi(\mathbf{y}|\mathbf{x},\theta) = \prod_{i=1}^{n_{obs}} \pi(y_i|x_i,\theta)$, so that y_i depends on only one x_i. The posterior marginals can then be written as

$$\begin{aligned} \pi(\theta_j|\mathbf{y}) &= \int \pi(\theta|\mathbf{y}) d\theta_{-j}, \\ \pi(x_i|\mathbf{y}) &= \int \pi(x_i|\mathbf{y},\theta)\pi(\theta|\mathbf{y}) d\theta, \end{aligned} \tag{3.22}$$

where x_i is the i:th component of $\mathbf{x}$, and θ_j is the j:th component of θ. Each of the densities will be approximated, letting $\tilde{\pi}$ be the approximation of π.

The numerical approximation of the latent field $\pi(x_i|\mathbf{y})$ is done in three steps; first the posterior $\pi(\theta|\mathbf{y})$ is approximated; second, if $\pi(\mathbf{y}|\mathbf{x},\theta)$ is non-gaussian, a Laplace approximation of $\pi(x_i|\mathbf{y},\theta)$ is computed; in the third step these two approximations are combined by numerical integration in (3.22).

Only a brief introduction is presented in this section, ignoring several features not relevant to our analysis. Interested readers can find more information about the inner workings of INLA in Rue et al. [15], and regarding the SPDE implementation in Lindgren et al. [11].

3.5.1 *Parameter estimation* $\tilde{\pi}(\theta|\mathbf{y})$

The approach to approximating $\tilde{\pi}(\theta|\mathbf{y})$, (3.20), is based on the law of total probability:

$$\begin{aligned} \pi(\mathbf{y}|\mathbf{x},\theta)\pi(\mathbf{x}|\theta) &= \pi(\mathbf{x},\mathbf{y}|\theta) = \pi(\mathbf{x}|\mathbf{y},\theta)\pi(\mathbf{y}|\theta) \\ \Rightarrow \pi(\mathbf{y}|\theta) &= \frac{\pi(\mathbf{x},\mathbf{y}|\theta)}{\pi(\mathbf{x}|\mathbf{y},\theta).} \end{aligned} \tag{3.23}$$

Inserting into (3.18), we obtain

$$\pi(\mathbf{y}|\theta) \propto \frac{\pi(\mathbf{x},\mathbf{y}|\theta)\pi(\theta)}{\pi(\mathbf{x}|\mathbf{y},\theta)} = \frac{\pi(\mathbf{x},\mathbf{y},\theta)}{\pi(\mathbf{x}|\mathbf{y},\theta).} \tag{3.24}$$

Using the calculated results, the approximation of the posterior distribution $\pi(\mathbf{y}|\theta)$ is done in INLA as

$$\tilde{\pi}(\theta|\mathbf{y}) \propto \left.\frac{\tilde{\pi}(\mathbf{y},\mathbf{x},\theta)}{\tilde{\pi}_G(\mathbf{x}|\mathbf{y},\theta)}\right|_{\mathbf{x}=\mathbf{x}^*(\theta)}, \tag{3.25}$$

where $\tilde{\pi}_G$ is a Gaussian approximation for the full conditional of $\mathbf{x}$, obtained as a taylor expansion of $\log\pi(\mathbf{x}|\mathbf{y},\theta)$ around the mode $\mathbf{x}^*(\theta)$

of the full conditional. For our model $\pi(\mathbf{y}|\mathbf{x},\theta)$ is Gaussian, and it follows that $\pi(\mathbf{x}|\mathbf{y},\theta)$ is also Gaussian, and no approximation will be necessary.

3.5.2 *Approximation of* $\pi(x_i|\mathbf{y},\theta)$

In our case $\pi(x_i|\mathbf{y},\theta)$ is Gaussian, and this step of INLA will not be necessary, otherwise the same approximation $\tilde{\pi}_G(\mathbf{x}|\mathbf{y},\theta)$ as in (3.25) can be used.

3.5.3 *Computing* $\pi(x_i|\mathbf{y})$

Having obtained approximations $\tilde{\pi}(\mathbf{x}|\mathbf{y},\theta)$ and $\tilde{\pi}(\theta|\mathbf{y})$, the posteriors are computed by numerical integration in (3.25) as:

$$\begin{aligned}\tilde{\pi}(\theta_j|\mathbf{y}) &= \int \tilde{\pi}(\theta|\mathbf{y})d\theta_{-j} \approx \sum_{\theta_{-j}} \tilde{\pi}(\theta|\mathbf{y})\Delta\theta_{-j},\\ \tilde{\pi}(x_i|\mathbf{y}) &= \int \tilde{\pi}(x_i|\mathbf{y},\theta)\tilde{\pi}(\theta|\mathbf{y})d\theta \approx \sum_{\theta} \tilde{\pi}(x_i|\mathbf{y},\theta)\tilde{\pi}(\theta|\mathbf{y})\Delta\theta.\end{aligned} \tag{3.26}$$

Part II

MODELLING

4 MODEL

This section will cover the implementation of mean and spatial components (as introduced in 3.2) of the model.

4.1 MEAN COMPONENT

4.1.1 *Covariates*

Erickson and Williams [4], found that elevation data is a significant predictor in a mean model. Using a similar model, the mean component will be modelled as:

$$\mu(\mathbf{u}) = \beta_0 + \beta_1 h(\mathbf{u}) + \beta_2 S_x(\mathbf{u}) + \beta_3 h(\mathbf{u}) S_x(\mathbf{u}) + \beta_4 S_x^2(\mathbf{u}), \quad (4.1)$$

where h, and S_x, are elevation and wind shelter of location $\mathbf{u}$; β_i are the weight of the covariates.

4.1.2 *Choice of wind direction and search distance*

Now consider the wind shelter data; as stated in section 2.4, the data was availible in eight different directions, with three different search distances for each direction. To decide which direction/search distance combination to use in the mean component, the mean model (4.1) was implemented in matlab, and linear regression (see a basic coursebook in statistics, e.g. Blom et al. [1]) is used to estimate the parameters. The parameters are then used to estimate the SWE data, and test which direction and distance yields the lowest Mean Squared Error (MSE).

Definition 11. *Mean Squared Error (MSE)*

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (\hat{x}_i - x_i)^2, \quad (4.2)$$

where $\hat{\mathbf{x}}$ is the estimation, and $\mathbf{x}$ the observed data [1].

The results of the regression test of wind shelter direction and search distance can be seen in Figure 12, which shows that index 6 (corresponds to wind direction 247.5°), with search distance of 250 yields the lowest MSE.

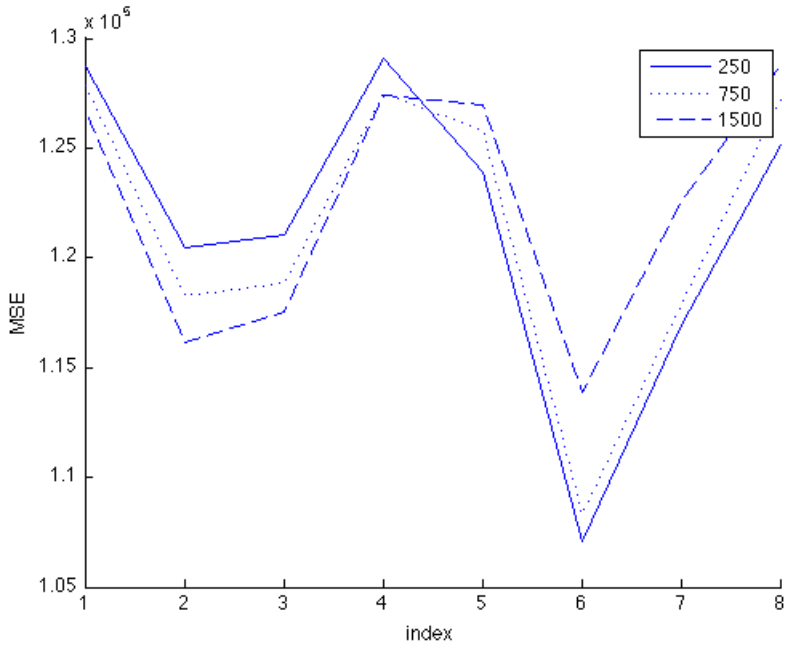


Figure 12: Regression test of wind direction and search distance combinations. Each point on the x-axis corresponds to an index in 22.5°, 67.5°, 112.5°, 157.5°, 202.5°, 247.5°, 292.5°, 337.5°.

4.2 SPATIAL COMPONENT

The spatial component is modelled using a GMRF as described in 3.3, where the latent field $\mathbf{x}$ will be assumed to be a GMRF to be able to estimate $\pi(\boldsymbol{\theta}|\mathbf{y})$ and $\pi(\mathbf{x}|\mathbf{y})$ by INLA. The neighbourhood of $\mathbf{x}$ is created by a triangulation, or mesh, function (as described in 3.3.3).

4.2.1 *Mesh*

The created triangulation can be seen in Figure 13. Vertices are placed at each measurement location (with some minimum distance between vertices added). The mesh is extended outside the area to reduce the effects of the boundrary conditions when the SPDE is solved; there is no practical interest in the outer extension, and the resolution does not have to be as fine as the inner area (Cameletti et al. [3]). The minimum angle of each triangle is specified to 21 degress, because of numerical stability issues.

4.3 FULL MODEL

The final model to be used with INLA can now be formulated as:

$$\mathbf{y} = \mathbf{A}(\boldsymbol{\eta} + 1 \cdot \beta_0) + \mathbf{B} \cdot \boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{4.3}$$

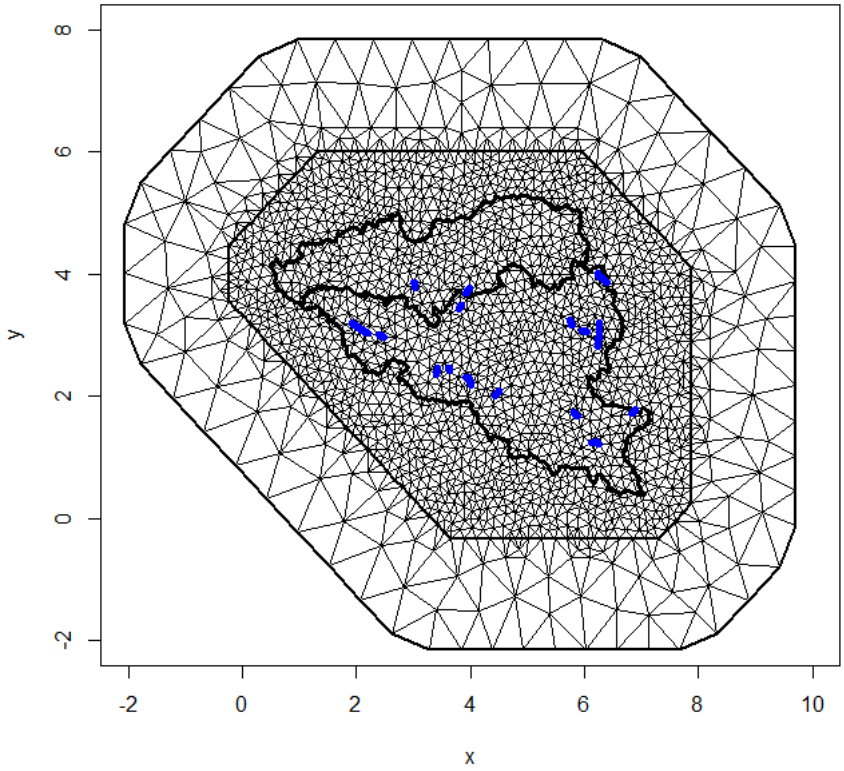


Figure 13: Mesh grid of the area.

where $(\boldsymbol{\eta} + 1 \cdot \beta_0)$ is the random spatial component and intercept, B is a matrix of covariates, and β the weights of the covariates. $\mathbf{A}$ is an matrix that maps the spatial component defined on the mesh to the observation locations, and ϵ is the nugget effect.

Comparing to (3.7), the latent field $\boldsymbol{x}$ can be formulated as $\boldsymbol{x} = \mathbf{A}(\boldsymbol{\eta} + 1 \cdot \beta_0) + \mathbf{B}\boldsymbol{\beta}$, with $\boldsymbol{\eta} \in (0, \mathbf{Q}^{-1})$ and $\boldsymbol{\mu} = \mathbf{A} \cdot 1 \cdot \beta_0 + \mathbf{B}\boldsymbol{\beta}$, giving $\boldsymbol{x} \in \mathcal{N}(\boldsymbol{\mu}, \mathbf{Q}^{-1})$. As a consequence, the observations can be written as $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\epsilon}$, where the nugget effect is $\epsilon \in \mathcal{N}(0, \mathbf{I}\sigma_\epsilon^2)$, resulting in $\boldsymbol{y}|\boldsymbol{x} \in \mathcal{N}(\boldsymbol{x}, \mathbf{I}\sigma_\epsilon^2)$.

The model is now finished, and estimations will be performed using the methods discussed in chapter 3. The R code for the estimation can be found in appendix A.

Part III

RESULTS

5 RESULTS

5.1 VALIDATION

The validation was done by removing one line (20 lines of observations exists in total) from the data to create data to perform validation on. Each line was tested by removing that line from the data set, and then running INLA on the remaining data. Almost every line caused the program to crash because failing to factorize $\mathbf{Q}$; only line 14 was successful.

The INLA estimation together with a linear regression estimation can be seen in Figure 14; residuals of the INLA estimation can be seen in Figure 16. MSE (4.2) is used, together with confidence coverage and bias, to evaluate the estimations

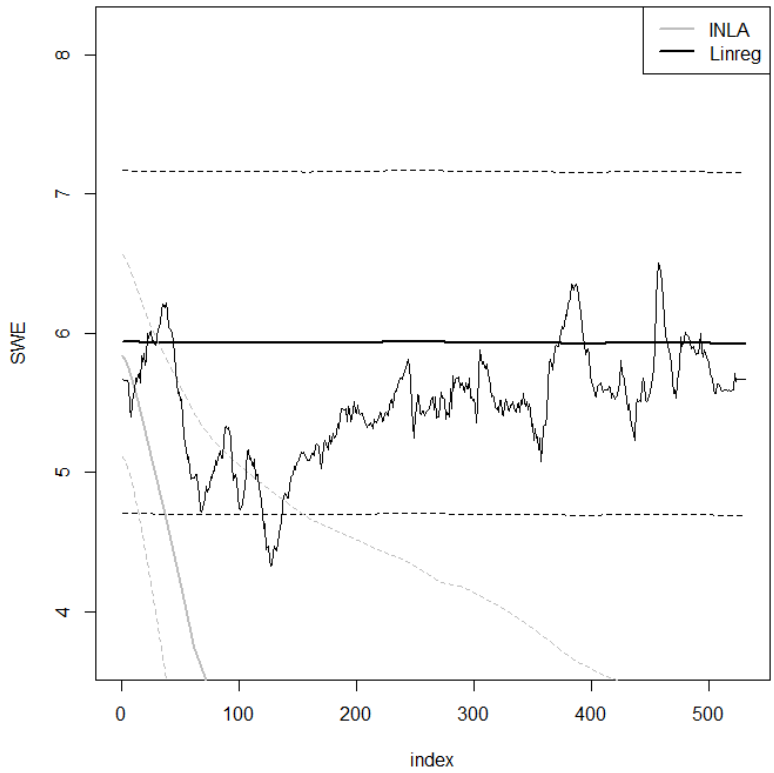


Figure 14: Validation plot of line 14. Line SWE data is plotted as a thin black line, the linear regression estimation is plotted in black, and its 95% confidence interval as a dotted black line. The INLA estimation is plotted in the same fashion, but in grey instead.

The figures showed that the INLA estimation was very poor; since most of the lines failed to be successfully estimated with INLA, the

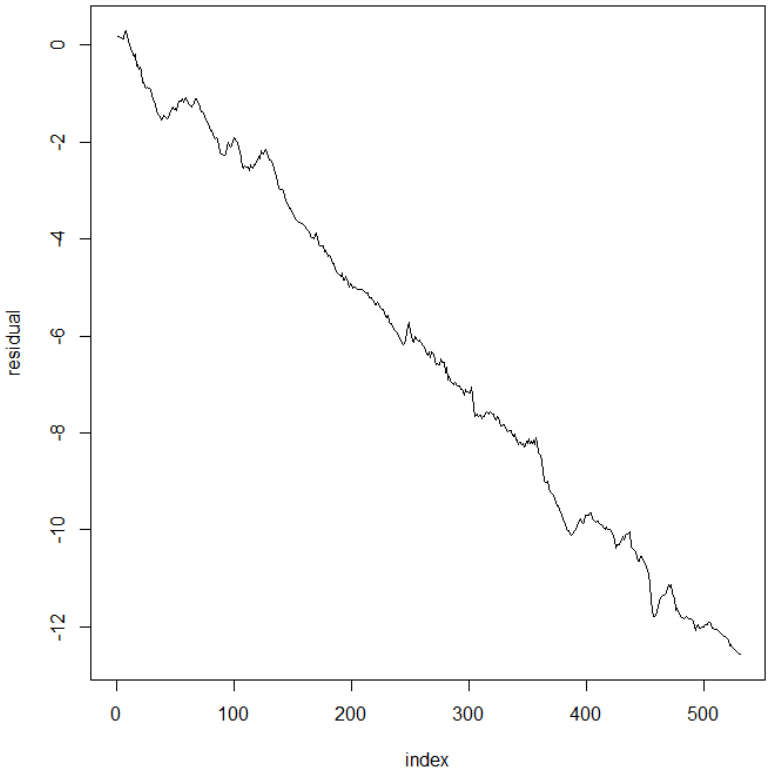


Figure 15: Residuals of the INLA estimation on line 14.

residuals of the linear regression is used with the autocorrelation function to check if dependence between the residuals exists.

The graph of the autocorrelation function can be seen in Figure 17, which shows a clear dependence between the residuals. An AR(1) component was added to the model, giving the extended model $\mathbf{y} = \mathbf{A}(\boldsymbol{\eta} + 1 \cdot \beta_0) + \mathbf{B}\boldsymbol{\beta} + \boldsymbol{\epsilon}_{\text{AR}(1)} + \boldsymbol{\epsilon}_{\text{nugget}}$, and the tests were redone.

INLA could, with the AR(1) component added, successfully run all tests; examples of estimations on line 14 and line 5 can be seen in Figure 19 and 18.

Line 14 had a MSE of $\sim$ 0.220 and its confidence interval covers $\sim$ 98.5% of the SWE data, it is a slight improvement over the linear regression which had a MSE of $\sim$ 0.35 and confidence interval coverage of $\sim$ 97.0%. Results of estimation on all the lines can be seen in Table 2.

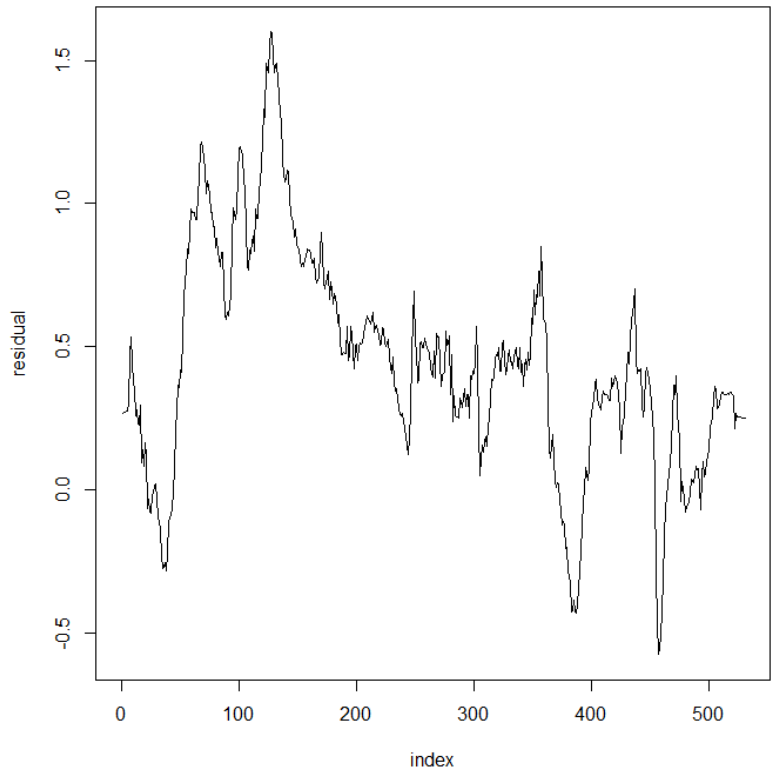


Figure 16: Residuals of the linear regression on line 14.

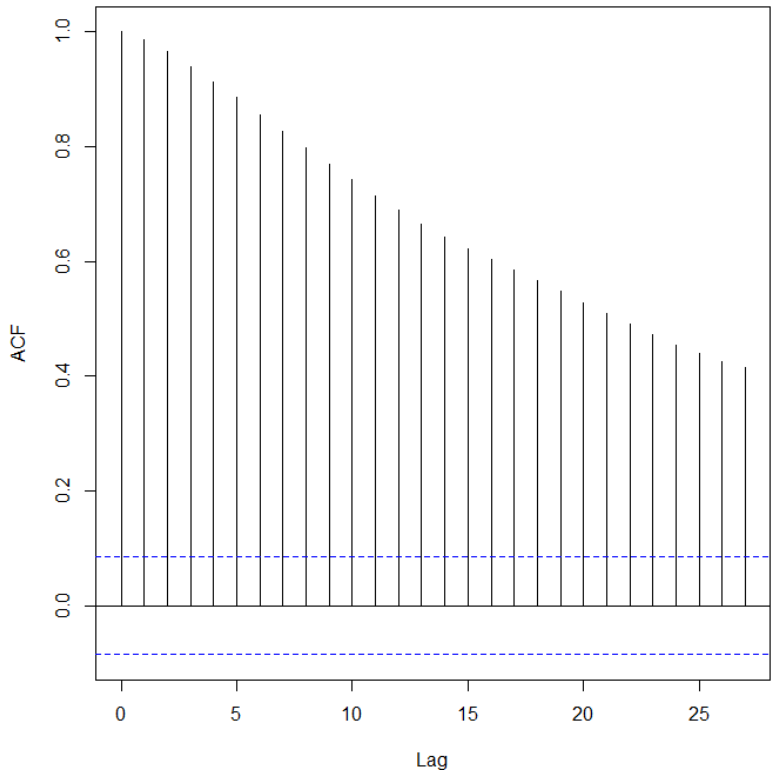


Figure 17: Autocorrelation function of the linear regression residuals.

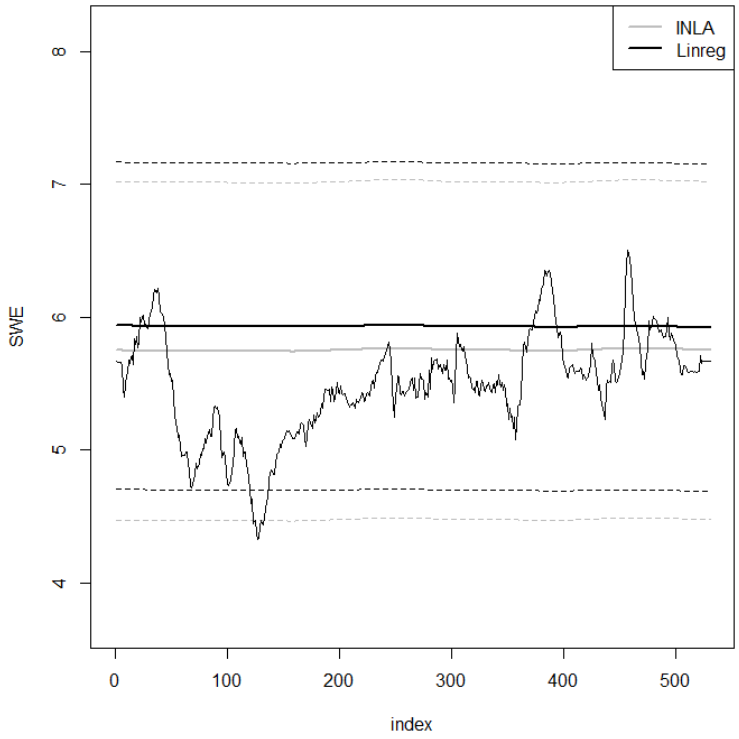


Figure 18: Validation plot of line 14. Line SWE data is plotted as a thin black line, the linear regression estimation is plotted in black, and its 95% confidence interval as a dotted black line. The INLA estimation is plotted in the same fashion, but in grey instead.

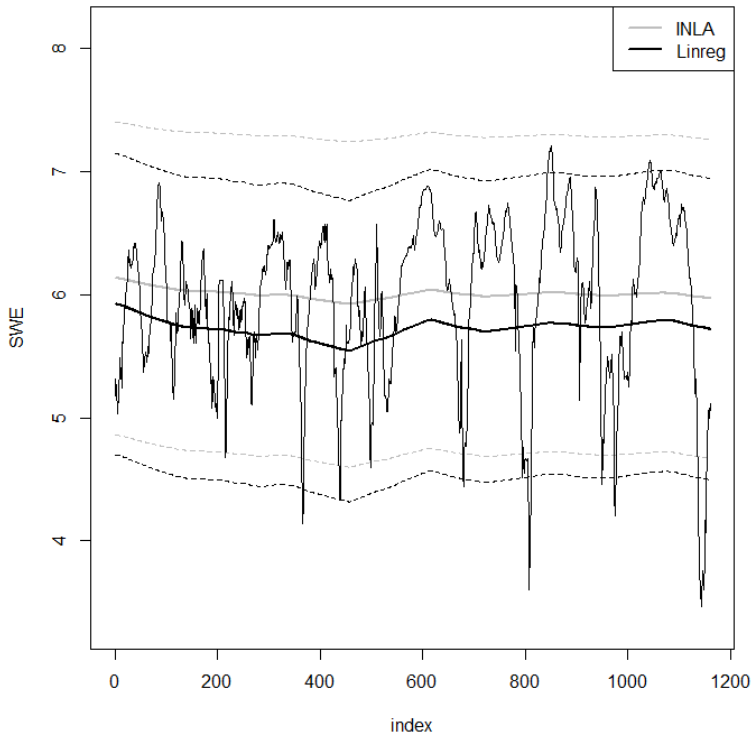


Figure 19: Validation plot of line 5.

	INLA		Linreg	
line	MSE	Conf.	MSE	Cov
1	0.48	94%	0.53	88%
2	0.26	99%	0.24	98%
3	0.13	99%	0.13	99%
4	0.44	95%	0.49	95%
5	0.76	87%	0.66	88%
6	0.045	100%	0.18	100%
7	0.47	93%	0.46	94%
8	0.28	100%	0.31	100%
9	0.090	100%	0.12	100%
10	0.81	90%	0.80	88%
11	0.032	100%	0.013	100%
12	0.86	89%	0.70	87%
13	0.22	98%	0.34	97%
14	0.029	100%	0.035	100%
15	0.84	84%	0.82	82%
16	0.64	88%	0.66	86%
17	0.25	99%	0.27	99%
18	0.16	100%	0.15	97%
19	0.29	100%	0.30	99%
20	0.42	96%	0.48	91%
Total	0.375	95.6%	0.384	94.4%

Table 2: Results of INLA estimations. `MSE` is the Mean Squared Error, `Cov` is the covariance coverage, and `Bias` is the bias of the estimator.

5.2 ESTIMATING SWE ON THE GRID

Normally, the estimation on the grid can be performed by INLA, but the availible computer (4 GB RAM) did not have enough memory to successfully run INLA with grid data added. This is circumvented by manually mapping the spatial component from the mesh to the grid by a suitable observation matrix, the mean component is also calculated for each grid point and added to the spatial component.

The mean component can be seen in Figure 20, and the spatial component in Figure 21. The estimations in the figures are shown in $\log(y+1)$, because the inverse transformation made the figures less disconcernable.

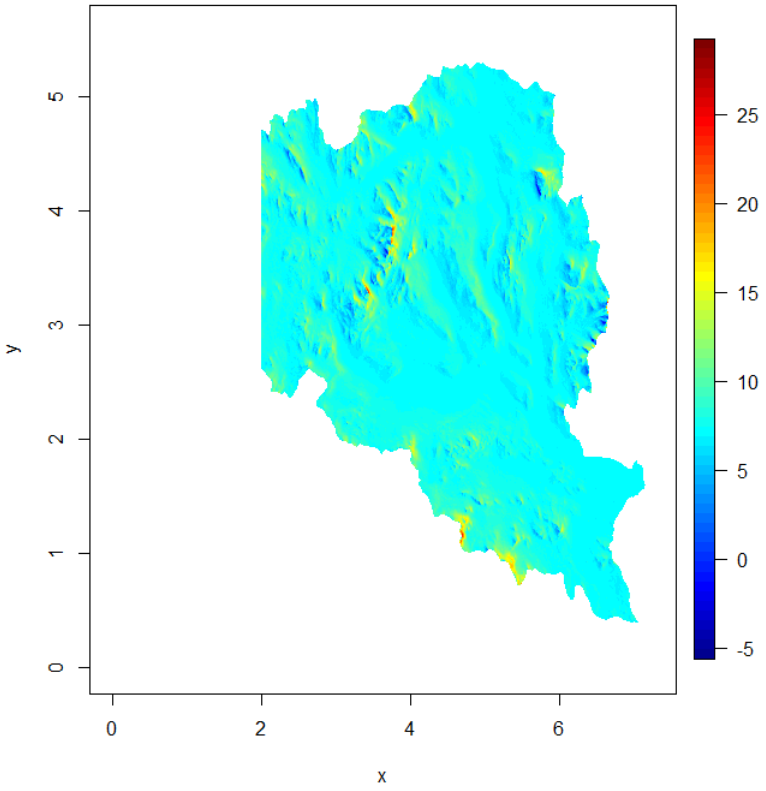


Figure 20: Mean component of the INLA estimation.

The full INLA estimation, where the mean and spatial components are added, can be seen in Figure 22; as a comparison, a linear regression estimation can be seen in Figure 23.

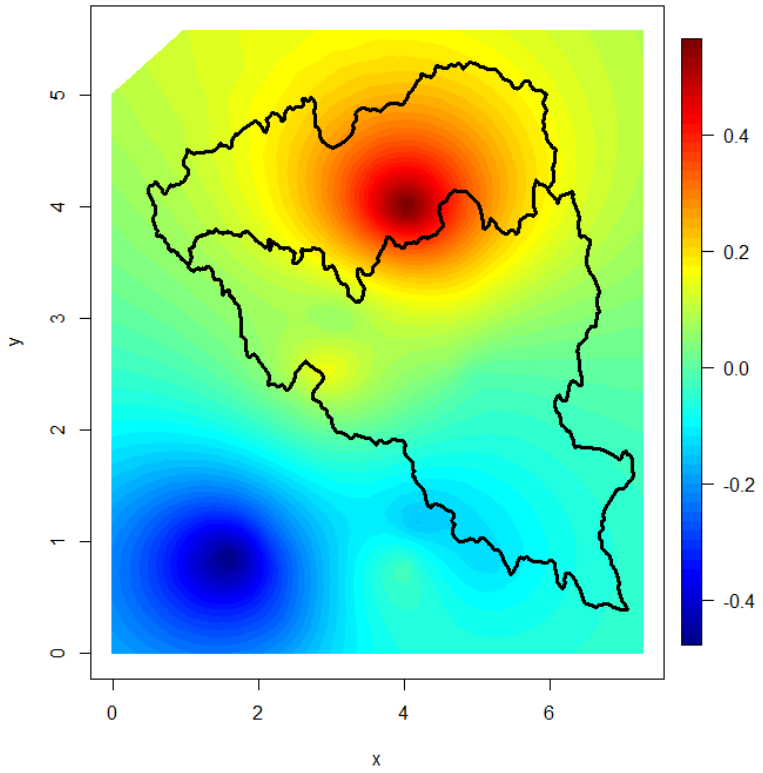


Figure 21: Spatial component of the INLA estimation.

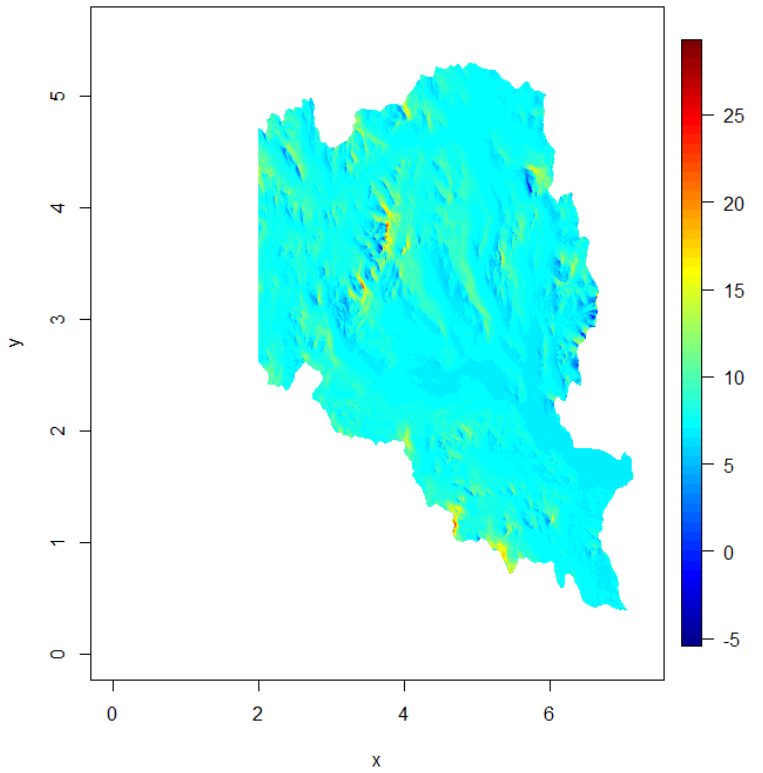


Figure 22: INLA estimation created by adding the mean and spatial component.

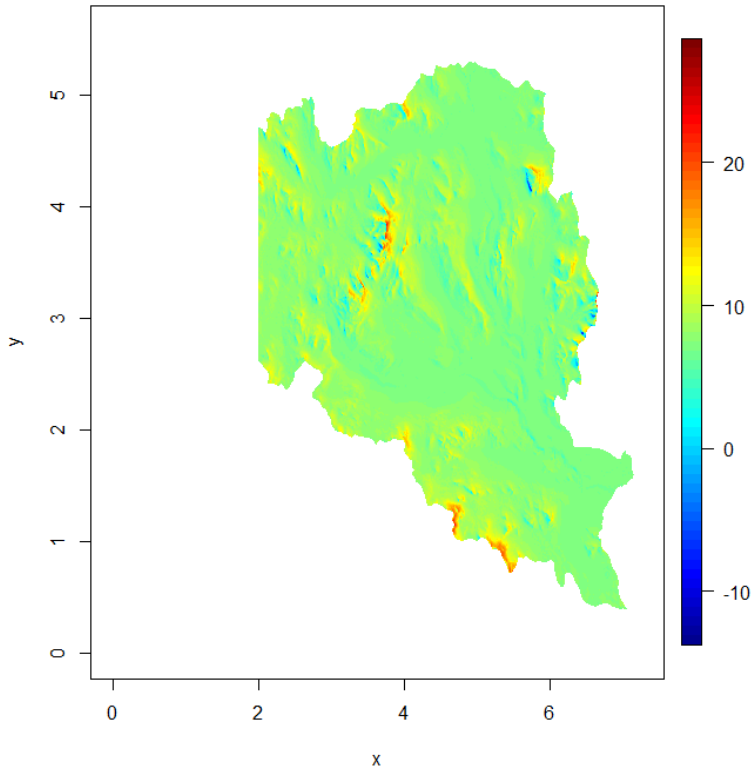


Figure 23: Linear regression estimation.

The estimation is unreasonably large at several locations at the map, the inverse transformation applied to a estimated value of 25 yields the SWE estimation of $e^{25} - 1$, a very large number. A likely cause of this is the mismatch of range between the covariates of observation locations and the grid points.

The covariates on the grid points are truncated, such that all covariate values outside the range of the observations is set to the largest or smallest value of the observations covariates. The results on the INLA and linear regression estimations can be seen in Figure 24 and 25.

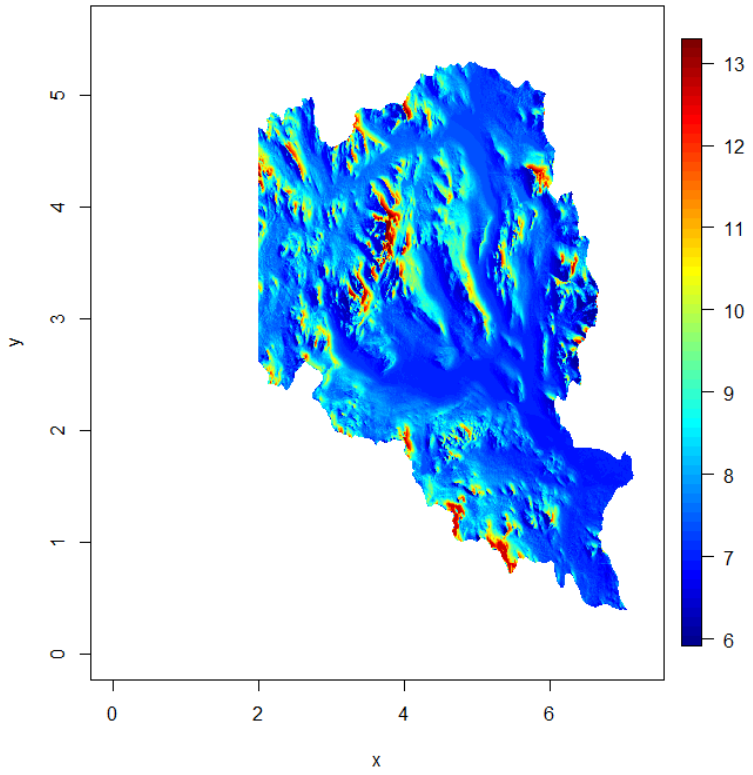


Figure 24: INLA estimation with truncated wind shelter and elevation data.

The INLA and Linear regression estimations are at some locations still very large, even with truncation. The maximum is about $e^{13} - 1$, and while not as large as $e^{25} - 1$, shows that there is some error in our estimation.

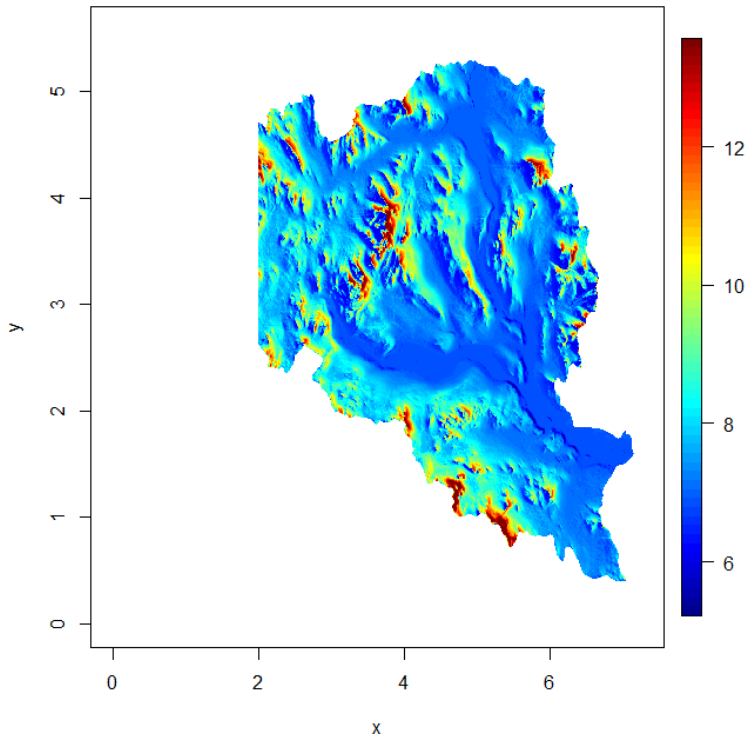


Figure 25: Linear regression estimation with truncated wind shelter and elevation data.

6 DISCUSSION

Validation data was created by removing one line from the data, INLA is performed on the rest of the data, and then tested by using the validation line.

The INLA estimation without the AR-part added could only be run with the line 14 used as validation without crashing due to factorization issues. The estimation for the line ended up being very poor as well. Using the residuals from the linear regression with the autocorrelation function, we saw a clear correlation between the residuals. An AR(1) component was added to the INLA code, and this made it possible to perform INLA with each line used for validation.

The results of the validation can be seen in table 2, where INLA had lower MSE than linear regression in 11 out of 20 cases. But the MSE in general between the validations were quite close, only differing by a small margin in all cases.

When the estimation was applied to the grid, we could see that for some locations on the map, the estimation was quite poor. SWE estimations of size $\sim e^{25}$ exists, which is a very strange result.

A major cause to this can be the mismatch of wind shelter and elevation data between the observation locations and the grid, where the range of the grid data is a magnitude longer than for the observations.

We tried to solve this by truncating both wind shelter and elevation data, but there is still severe overestimation at some locations. This is probably due to the size of the intercept ($\beta_0 = 5.58$), the covariates multiplied with its corresponding weight does not yield negative values to offset the high value of the intercept, resulting in overestimations.

As we can see in Figure 21, the spatial component at most locations is quite small compared to the mean component. One interesting question is whether we have enough observations to form a spatial dependence relevant to the estimations.

On a final note, INLA did not manage to produce results significantly better than the linear regression in the validation, and provided the same overestimations as the linear regression when applied to the grid data.

6.1 FUTURE WORK

Since the computer used in the estimations were did not have enough memory. One improvement in the existing code could be using a more powerful computer to run the INLA estimations, which makes it possible to include the grid in the model fitting done by INLA.

One big issue in the estimations are the extreme points where the estimated SWE values are either unreasonable big or negative. We tried to solve it by truncating the data; while we could always wish for observations at these locations, a solution could be trying extrapolate data using e.g. a spline function.

Another solution to the problem with the wind shelter range could be creating a simpler mean model, where the wind shelter is removed.

A final suggestion regarding the modelling approach, is modelling the snow depth and snow density separately, and then combine the estimations to estimate the SWE.

Part IV

APPENDIX

A

CODE

```
# Perform the INLA operations on the data to check the valdiation on line R
source("misc.r")

inla.AR = function(data, n)
{
# Check if snow data set is supplied
if ( missing(data) ) {
stop("Error, data sets is required.")
}

if ( missing(n)  ) {
stop("Validation line must be specified")
}

# Perform inla
res <- perform.inla(data = data, line = n)

return(res)
}

perform.inla = function(data, line)
{

# Create one dataset for line n
data.val <- create.val(data,line)
data.est <- remove.val(data,data.val)

# create mesh
mesh <- inla.mesh.create.helper(
# Locations that should be present
points = cbind(data.est$x,data.est$y),
# Determines the domain extent
points.domain = map,
# Allow no smaller triangle than the length below
cutoff = 0.05,
# Maximum length of the edge, first parameter
# is inside the domain, latter is outside.
max.edge = c(0.25,1),
# Offsets outside the edge, the negative values
# gives a scaled offset.
offset = c(-.1,-.2),
```

```
# Minimum angle of triangles
min.angle = 21
)

# Saving position for vertrices
field.indicies <- inla.spde.make.index("field", n.mesh = mesh$n)

# Creating the SPDE model
spde <- inla.spde2.matern(mesh = mesh,
# Operator order
alpha = 2,
# Setting up prior variance
prior.variance.nominal = 1,
# Precision on observations
theta.prior.prec = 1
)

# --------------  Creating data stacks  ----------------------
est.AR <- make.AR(data.est)
val.AR <- make.AR(data.val)

# Estimation stacks
A.est <- inla.spde.make.A(mesh = mesh, loc = cbind(data.est$x,data.est$y))
stack.est <- inla.stack(data = list(SWE = data.est$SWE),
A = list(A.est,1),
effects = list(c(field.indicies, list(Intercept = 1)),
list(Elevation = data.est$elevation,
Windshelter = data.est$windshelter,
Windshelter2 = data.est$windshelter2,
index = est.AR$index, # Added for AR(1)
replicate = est.AR$repl
))
,tag = "est"
)



# Validation stack
A.val <- inla.spde.make.A(mesh = mesh, loc = cbind(data.val$x,data.val$y))
stack.val <- inla.stack(data = list(SWE = NA),
A = list(A.val,1),
effects = list(c(field.indicies, list(Intercept = 1)),
list(Elevation = data.val$elevation,
Windshelter = data.val$windshelter,
Windshelter2 = data.val$windshelter2,
index = val.AR$index, # Added for AR(1)
```

```
replicate = (val.AR$repl + 20)
))
,tag = "val"
)

#Prediction stack
stack.pred <- inla.stack(data = list(SWE = NA),
A = list(1),
effects = list(c(field.indicies, list(Intercept = 1))
),
tag = "pred"
)

# Putting together the stacks
stack <- inla.stack(stack.est, stack.val, stack.pred)

# Setting up the formula
formula <- SWE ~ -1 + Intercept + Elevation*Windshelter +
Windshelter2 + f(field, model = spde, initial = 0) +
# f(replicate, model = "iid"), part of the non-AR model
f(index, model = "ar1", replicate = replicate)

# Setting up initial theta, not used in the non-AR model
a<-inla.models()$likelihood$gaussian$hyper
a$theta$initial <- 15
a$theta$fixed <- TRUE

# Perform inla
r <- inla(formula, data = inla.stack.data(stack,spde = spde),
family = "gaussian",
control.predictor = list(A = inla.stack.A(stack), compute = TRUE),
control.inla = list(h = 0.001),
verbose = TRUE,
control.family = list(hyper = a),
num.threads = 1
)

res <- list(r = r, stack = stack, mesh = mesh, n = data.val$line[1],
            size = tail(val.AR$index,1) )

return(res)
}
```

BIBLIOGRAPHY

[1] Gunnar Blom, Jan Enger, Gunnar Englund, Jan Grandell, and Lars Holst. *Sannolikhetsteori och statistikteori med tillämpningar*. Studentlitteratur, 5:5 edition, 2005.

[2] G.E.P. Box and D.R. Cox. An analysis of transformations. *Journal of the Royal Statistical Society*, 26(2):211–252, 1964.

[3] Michela Cameletti, Finn Lindgren, Daniel Simpson, and Håvard Rue. Spatio-temporal modeling of particulate matter concentration through the spde approach. http://www.math.ntnu.no/~daniesi/Cameletti_et_al_submitted.pdf, 2012. Accessed 07/03/2013.

[4] Tyler A. Erickson and Mark W. Williams. Persistence of topographic controls on the spatial distribution of snow in rugged mountain terrain, Colorado, United States. *Water Resources Research*, 41:1–17, 2005.

[5] Gerald Farin. *Curves and Surfaces for CAGD - A Practical Guide*. Morgan Kaufmann Publishers, fifth edition edition, 2002.

[6] Alan E. Gelfand, Peter Diggle, Peter Guttorp, and Montserrat Fuentes, editors. *Handbook of Spatial Statistics*. Chapman & Hall/CRC, 2010.

[7] C.A. Gill and D.N. Joanes. Comparing measures of sample skewness and kurtosis. *The Statistician*, 1(47):183–189, 1998.

[8] Helge Holden, Bernt Øksendal, Jan Ubøe, and Tusheng Zhang. *Stochastic Partial Differential Equations : A Modeling, White Noise Functional Approach*. Springer Science + Business Media, second edition edition, 2010.

[9] Theo Kypraios. Introduction to Bayesian Statistics. http://www.maths.nottingham.ac.uk/personal/tk/files/talks/Cran_10_12.pdf. Accessed 17/01/2013.

[10] Finn Lindgren. *Image Modelling and Estimation - A statistical approach*. Third edition edition, 2006.

[11] Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of Royal Statistical Society*, 4(73):423–498, 2011.

[12] Henrik Madsen. *Time Series Analysis*. Chapman & Hall/CRC, first edition edition, 2008.

[13] Dean Morel. What is Skew and Why is it Important. http://www.fusioninvesting.com/2010/09/what-is-skew-and-why-is-it-important/, 2012. Accessed 15/02/2013.

[14] NASA. Snow Water Equivalent. http://disc.gsfc.nasa.gov/hydrology/data-holdings/parameters/snow_water_equivalent.shtml. Accessed 15/01/2013.

[15] Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of Royal Statistical Society*, 2(71):319–392, 2009.

[16] Martin Sköld. *Computer Intensive Statistical Methods*. Mathematical Statistics, Lund University, Box 118, SE-221 00 Lund, Sweden, http://www.maths.lth.se/, second printing edition, 2006.

[17] Peter Whittle. On stationary processes in the plane. *Biometrika*, 41:434–449, 1954.

[18] Peter Whittle. Stochastic processes in several dimensions. *BofIntStatInst*, 40:974–994, 1963.

[19] Adam Winstrahl, Kelly Elder, and Robert E. Davis. Spatial Snow Modeling of Wind-Redistributed Snow Using Terrain-Based Parameters. *Journal of Hydrometeorlogy*, 3:524–538, 2002.

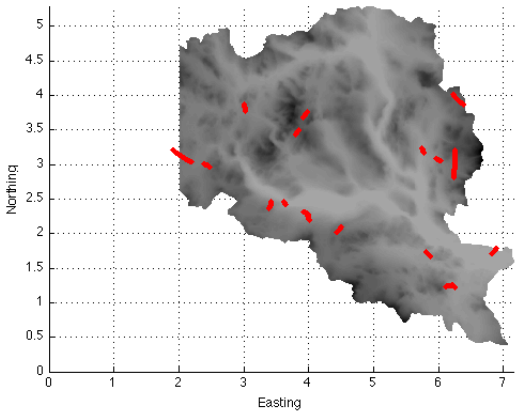
Estimering av snödjup med spatial statistik

Magnus Öhlund $\pi 07$

5 april 2013

Vi är intresserade av att estimera snödjupet i ett område för att mäta hur mycket vatten som kommer flöda in i floder och vattencisterner när snön smälter på våren. Området som är i fokus i detta projekt är Kultsjön/Rensaren i norra Sverige, där vi har mätningar av snödjup längs linjer.

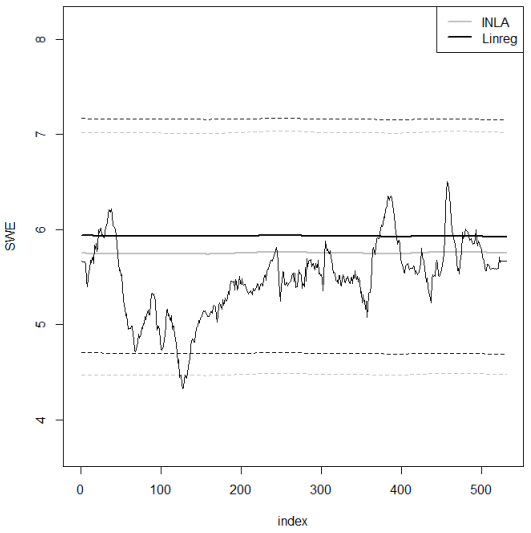
Dessa mätningar är gjorda med hjälp av ultraljud, där det resulterande snödjupet är beräknat på hur lång tid det tar för signalen att komma tillbaka till mätutrustningen. Utöver snölinjerna, vet vi även höjden på området, samt hur utsatt för vind alla platser på området är.



Figur 1: Karta över Kultsjön/Rensaren, de röda linjerna är mätningarna av snödjup.

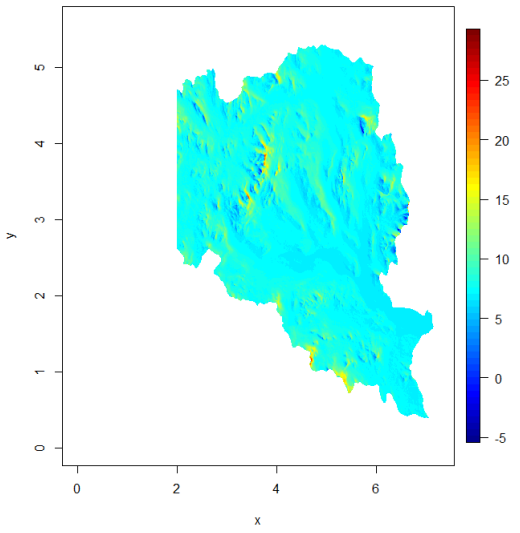
Närliggande platser på ett område kan vara beroende av varandra, och med spatial statistik försöker vi skapa en statistisk modell som fångar detta beroende. De tillgängliga snölinjerna modelleras med hjälp av spatial statistik, där höjddatan och exponeringen används som förklarande variablar i skattningen.

När vi konstruerat en statistisk modell, är nästa steg att validera modellen genom att ta bort en av de tillgängliga linjerna och sedan rekonstruera denna med hjälp av enbart mätningar längs de återstående linjerna; efter valideringen appliceras modellen på området. I valideringen så jämför vi vår modell mot en modell skapad med linjär regression (en grundläggande statistisk metod för att anpassa en ekvation till data).



Figur 2: Exempel på en återskapad linje, den grå linjen är den framtagna statistiska modellen. Den tunna svarta linjen är observationer längs den borttagna linjen, och den svarta linjen är rekonstrueringen med linjär regression.

Alla linjer återskapas, och generellt sett så gav modellen med spatial statistik något bättre resultat än linjär regression. Då modellen gav vettiga resultat i valideringen, så applicerar vi den på helan området; resultatet ses i figuren nedan.



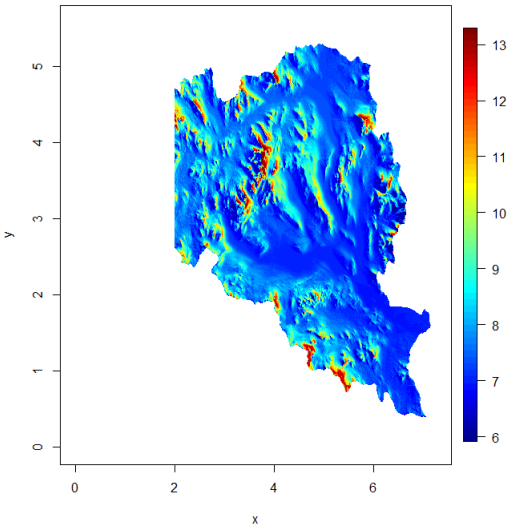
Figur 3: Estimering av snödjupet på området. Figuren är i *logaritmisk* skala.

Vi kan se i figuren att vi har toppar som är $2.72^{25} \approx 72\ 004\ 899\ 337$, vilket är ett väldigt orimligt resultat. Ett problem med de förklarande variablarna, höjd och exponering, är att linjernas värden hos de förklarande variablarna inte alls täcker områdets värden på de förklarande variablarna.

En enkel metod för att försöka lösa problemet med de förklarande variablarna är att trunkera höjd- och exponeringsdatan, så att vi trunkerar alla värden som över- eller understiger värden på linjernas; d.v.s. dessa värden sätts till de högsta eller lägsta värde längs med linjerna.

Estimeringen görs om med de trunkerade förklarande variablarna, de nya resultaten ses i figur 4.

Här ser resultatet mer rimligt ut, dock så



Figur 4: Ny estimering av snödjupet på området, de förklarande variablarna är trunkerade.

är estimeringen för vissa platser fortfarande stora, topparna är på $2.72^{13} \approx 442\ 413$, vilket är för stort för att vara ett rimligt resultat.

Förslag till framtida arbeten för att lösa problemet med för stora värden, kan vara t.ex. en smartare lösning till de förklaranade variablarna; trunkeringen löste problemet till viss del, men resultatet blev inte riktigt bra. Mätdata kring topparna där estimeringarna är stora hade också hjälpt, då ytterligare data förhoppningsvis hade förbättrat modellen och, på så sätt, tagit hand om problemet.

Master's Theses in Mathematical Sciences 2013:E19
ISSN 1404-6342

LUTFMS-3210-2013

Mathematical Statistics
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/